Supply auditing: what it is (and isn't)

MoneroTopia Miami 2022

7 April 2022

Disclaimer

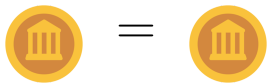
The material in this presentation is for informational and educational purposes only.

It is not financial, investment, or other professional advice, and should not be interpreted as such.

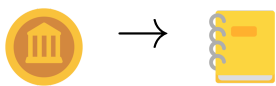
What is inflation?

In digital assets, we need to ensure that value **able to be spent** is correct according to protocol rules.

- Transfer transactions must not create more value than they uniquely consume.



- Value-adding transactions must not create more value than expected.



The big questions

- In what ways can inflation occur?
- How certain can we be that inflation has not occurred?
- What are the consequences if inflation does occur?

It's subtle.

Statements that are wrong

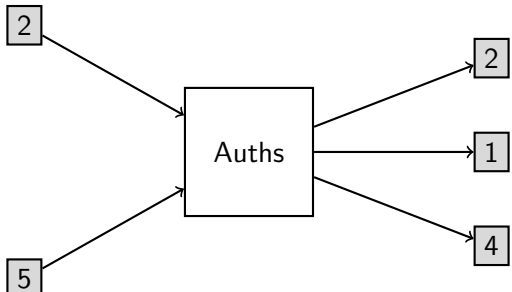
The following statements are incorrect (but also subtle):

Inflation can't occur in a transparent asset.

Inflation can't occur if you count coinbases properly.

Inflation can't occur if you use transparent migrations.

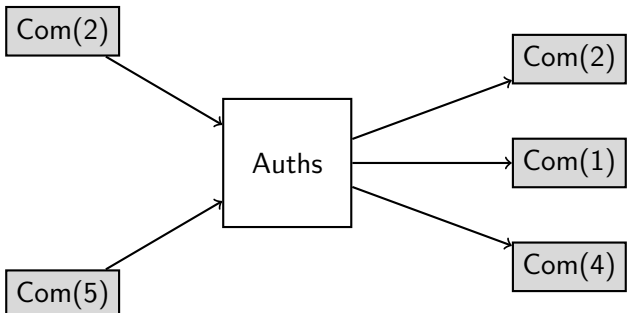
In **transparent assets**, the amounts consumed and generated in transactions are in the clear, typically with straightforward authorizations.



We can look at input and outputs amounts, and assert that they balance:

$$2 + 5 - (2 + 1 + 4) = 0$$

In **opaque assets**, the amounts consumed and generated in transactions are hidden using one-way commitments, typically with more complex authorizations.



We can't see the amounts, but the following arithmetic holds:

$$\mathrm{Com}(2) + \mathrm{Com}(5) - (\mathrm{Com}(2) + \mathrm{Com}(1) + \mathrm{Com}(4)) = \mathrm{Com}(0)$$

We need to ensure:

That value in transactions is hidden.*

That unexpected value cannot be created.

That value in transactions balances.

That funds cannot be spent multiple times.

That authorizations are correct.

These come with certain cryptographic and implementation risks.

*Opaque only!

Case study: Monero

Like other opaque assets, Monero uses a **linking tag** to prevent double-spend attempts.

Each consumed coin has a revealed tag that does not identify the coin directly; if a tag has been revealed before, the transaction is rejected.

A bug existed that would allow the owner of a coin to reveal multiple linking tags for the coin, effectively octo-spending it.

Case study: Monero

Mitigation: The bug was privately disclosed and patched in a client update, with details revealed after the update.

Effect: The bug was definitively not exploited. Evil tags have an algebraic property that can be checked. No tags had this property.

Case study: Zcash

Zcash proving system security depends in part on multi-party ceremonies that generated parameters to bootstrap its Sprout and Sapling protocols.

The ceremonies came equipped with **transcripts** that allowed third parties to check the validity of certain operations.

A problem with the Sprout transcript existed that would allow an observer to generate transactions adding arbitrary hidden value to the Sprout pool.

Case study: Zcash

Mitigation: The problem was privately disclosed and patched using a proving system and protocol upgrade, with details revealed after the upgrade.

Effect: It is not possible to definitively know if the problem was exploited.

Pretend case study: fancy adversaries

Balance protection in opaque assets that use commitments (like Monero and Zcash) depends in part on commitments being bound to unique values:

$$\mathrm{Com}(3) \neq \mathrm{Com}(1003)$$

Revealing a commitment opening to a different value would fool balance checks.

A fancy adversary who learns an algebraic relationship between certain parameters could do this; but it's considered **exceptionally unlikely** to be feasible without major computing advances.

Pretend case study: fancy adversaries

Mitigation: Some options include using different types of cryptographic plumbing to build protocols, but this has other cryptographic consequences.

Effect: It would not be possible to definitively know if the problem were exploited.

Why not...

Why can't we just count the coinbase funds added to opaque pools?

Higher-than-expected coinbase value addition is one avenue for inflation to occur, but it is not sufficient to check this.

This wouldn't detect:

- Problems with authorizations (Monero, Zcash)
- Cryptographic breaks (fancy adversary)
- Other consensus rule issues (Bitcoin)

Why not...

Why can't we use transparent migrations to ensure too much value doesn't come out?

Zcash used a transparent migration requiring opaque funds leaving the Sprout pool to the newer Sapling pool to reveal the amounts.

If more funds leave Sprout than were transparently put in, all remaining opaque funds would be frozen and not allowed to leave.

Other protocols could choose to take different actions in response to a negative pool trigger.

Why not...

Consequences of this:

- If an attacker pulled out enough funds to make the pool negative, any remaining honest users' funds would be subject to protocol-specific consequences.
- An attacker could still pull out inflated funds, being careful not to bring the pool negative.
- An attacker could still spend inflated funds within the pool.
- Migration transactions reveal amounts and contain metadata.

What about transparent assets?

So there's no inflation in transparent assets, right?

This is not the case.

Transparent assets can have (and have had) inflation bugs, some of which have been exploited.

Case study: Bitcoin

The Bitcoin protocol contains consensus rules and code requiring consumed and generated coin amounts to balance.

A bug existed that allowed coins to be generated with such large amounts that when summed, the total would overflow and fool the balance check.

Case study: Bitcoin

Mitigation: An exploit was observed and patched in a client update.

Effect: The bug was definitively exploited, and the chain was forked to undo the exploited transaction.

Lessons

There are many things to take away from these examples (and there are others).

- Not all forms of inflation are the same. They could arise from **cryptographic breaks**, **bad consensus rules**, **implementation flaws**, or **bad protocols**.
- Inflation in opaque assets may be detectable, but not always.
- Inflation can occur in transparent assets; while it is detectable, it has consequences.
- Techniques like transparent migrations are not sufficient to mitigate all possible effects of opaque-asset inflation.

Questions